

## Guión 3

### *Introducción a la compilación y ejecución de programas Java mediante J2SE y Netbeans*

*Noviembre de 2006*



**DECSAI**  
Departamento de Ciencias  
de la Computación e I.A.  
Universidad de Granada

# Nuevas Tecnologías de la Programación

Curso 2006/2007



# Índice

<b>1. Introducción</b>	<b>5</b>
<b>2. Utilización de j2sdk</b>	<b>5</b>
2.1. Compilación de programas . . . . .	6
2.2. Ejecución de programas . . . . .	6
2.3. Compilación y ejecución del programa serpiente . . . . .	7
<b>3. Introducción al uso de netbeans</b>	<b>7</b>
3.1. Importar código Java en netbeans . . . . .	8
3.1.1. Programa Hello World . . . . .	9
3.1.2. Programa de la serpiente . . . . .	12
3.2. Importing Existing Java Source Code Into NetBeans IDE 5.5	14
<b>4. Tutoriales en la página de Netbeans</b>	<b>14</b>
4.1. NetBeans IDE 5.5 Quick Start Guide . . . . .	14
4.2. Introduction to Developing General Java Applications . . . . .	14



## 1. Introducción

Para trabajar en Java usaremos la versión 1.5.0 o bien la 1.6.0 del j2sdk (también conocida como J2SE 5.0). Este software puede encontrarse en la página web de Sun <http://java.sun.com>. De esta página podemos descargarnos el fichero `jdk-1_5_0_09-linux-i586.bin` que contiene el j2sdk 1.5.0 o el fichero `jdk-6-rc-linux-i586.bin` que contiene la versión 1.6.0

Además usaremos el interfaz de desarrollo de software **netbeans** version 5.5 (fichero `netbeans-5_5-linux.bin`) que puede descargarse de la página <http://www.netbeans.org>. En la página de Sun (<http://java.sun.com>) existe también una versión de j2sdk 5.0 con el netbeans 5.5 ya incluido.

En los ordenadores de las aulas de prácticas de la ETSIIT ya está instalado el anterior software en los siguientes directorios:

- `/usr/local` : Directorio raíz dónde se encuentran las versiones 1.5.0 y 1.6.0 del j2sdk y el netbeans 5.5
- `/usr/local/jdk1.5.0_09`: Contiene el j2sdk 1.5.0 . Dentro del subdirectorio `bin` podemos encontrar los programas `java`, `javac`, `javadoc`, etc.
- `/usr/local/jdk1.6.0`: Contiene el j2sdk 1.6.0. Dentro del subdirectorio `bin` podemos encontrar los programas `java`, `javac`, `javadoc`, etc.
- `/usr/local/netbeans-5.5`: Contiene el netbeans 5.5. Dentro del subdirectorio `bin` podemos encontrar el binario del programa `netbeans`.

## 2. Utilización de j2sdk

Este paquete de programas nos permite entre otras cosas, compilar programas en Java, ejecutarlos, depurarlos, generar documentación html a partir de los comentarios de los códigos fuente, etc. Cada una de estas acciones se realizan mediante un determinado programa del paquete j2sdk. Como hemos dicho, los programas de j2sdk cuelgan de la carpeta `/usr/local/jdk1.5.0_09` o `/usr/local/jdk1.6.0`. Concretamente los ejecutables están en las carpetas `/usr/local/jdk1.5.0_09/bin` o `/usr/local/jdk1.6.0/bin`. Estos programas se pueden invocar indicando la ruta completa del directorio donde se encuentran. Por ejemplo mediante:

```
/usr/local/jdk1.6.0/bin/javac Prog.java
```

O bien podemos invocarlos indicando el programa a ejecutar pero sin hacer explícita su ubicación. Por ejemplo:

```
javac Prog.java
```

Para operar de esta segunda forma, el sistema operativo debe conocer el directorio donde está el programa a ejecutar. Una forma de hacerlo permanentemente es añadiendo dicho directorio a la variable PATH dentro del fichero de inicio correspondiente:

- Si usas una C shell (*cs*h o *tc*sh), edita el fichero `~/ .cshrc` y añade:

```
setenv PATH /usr/local/jdk1.6.0/bin:${PATH}
```

- Si usas *ksh*, *bash* o *sh*, edita el fichero `~/ .profile` y añade:

```
PATH=/usr/local/jdk1.6.0/bin:$PATH
export $PATH
```

Por último, carga de nuevo el fichero correspondiente (en la C shell ejecuta: `source ~/ .cshrc`, y en el resto ejecuta: `$HOME/.profile`) y comprueba la corrección de la operación con el comando `which javac`.

## 2.1. Compilación de programas

Vamos a compilar por ejemplo el programa `Example.java` que se encuentra en el directorio `/fenix/depar/ccia/ntp/Java/Sources/P1`.

Debemos copiar este fichero en uno de nuestros directorios de nuestra cuenta. Luego nos cambiamos a ese directorio y ejecutamos la orden:

```
javac Example.java
```

Como resultado hemos obtenido el fichero `Example.class` en el mismo directorio, que corresponde al código bytecode del fichero `Example.java`.

## 2.2. Ejecución de programas

Una vez compilado el programa en java, podemos ejecutarlo ahora con la siguiente orden:

```
java Example
```

## 2.3. Compilación y ejecución del programa serpiente

En el siguiente directorio puedes encontrar varios ficheros en java que implementan el juego de la serpiente contruidos con el paquete de clases AWT:

```
/fenix/depar/ccia/ntp/Java/Sources/serpienteAWT/serpiente
```

Copiad el directorio completo a vuestra cuenta. Cuando no hay ningún fichero compilado, podemos compilarlos todos, al compilar la clase que contiene la función `main()`. Colocándote en el directorio padre del directorio `serpiente` de tu cuenta, ejecuta la siguiente orden para compilar todos los ficheros:

```
javac serpiente/Main.java
```

Ahora puedes ejecutar el programa con la orden:

```
java serpiente/Main
```

En la anterior orden, también es posible usar `serpiente.Main` en lugar de `serpiente/Main`.

## 3. Introducción al uso de netbeans

Netbeans es un entorno de desarrollo integrado (IDE) que permite editar programas en java, compilarlos, ejecutarlos, depurarlos, construir rápidamente el interfaz gráfico de una aplicación eligiendo los componentes de una paleta, etc.

Para ejecutar este entorno debemos ejecutar la orden:

```
netbeans
```

Esta orden ejecuta el netbeans usando `jdk 1.6.0` como entorno para compilar y ejecutar los programas Java que construyamos, pues así se eligió al instalar netbeans en los ordenadores de las aulas de prácticas. Si queremos usar `jdk 1.5.0` ejecutaríamos la orden:

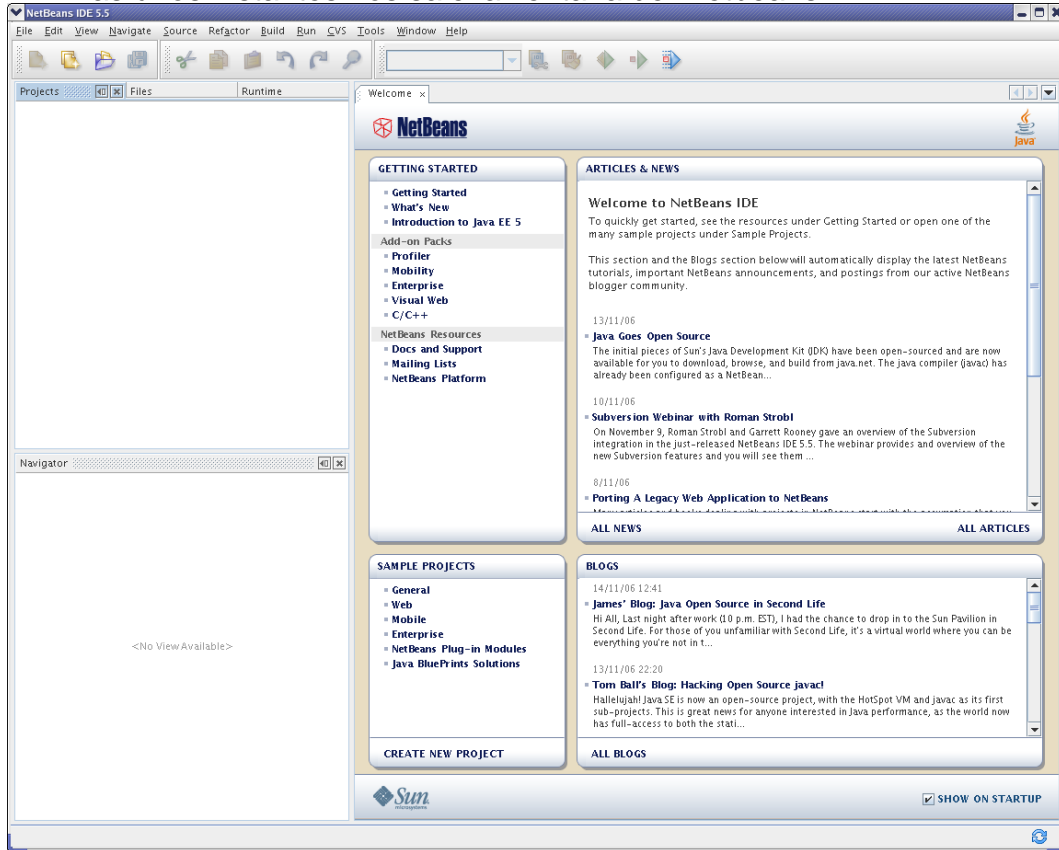
```
netbeans --jdkhome /usr/local/jdk1.5.0_09
```

La primera vez que ejecutemos el entorno, nos sale una ventana que pregunta si hay otras versiones de Netbeans instaladas en el sistema. Debemos contestar que no. Esta primera ejecución de Netbeans crea en nuestro `home` un directorio `.netbeans/5.5`. Este directorio puede llegar a ocupar alrededor de 30 Megabytes. Si no dispones en tu cuenta de ese espacio libre, puedes ejecutar netbeans de la siguiente forma para que el directorio `.netbeans/5.5` se cree en `/tmp`:

```
netbeans --userdir /tmp/netbeans
```

Esta última forma tiene el inconveniente que al apagar el ordenador perderemos nuestra configuración personal del entorno Netbeans, ya que el directorio `tmp` se pierde.

Tras unos instantes nos sale la ventana del Netbeans.



En la parte derecha del IDE hay unos enlaces que nos permiten entrar en tutoriales de inicio que conviene que les echemos un vistazo. Por ejemplo pinchando en el enlace *Start Guide* accedemos a una página en la que podemos encontrar los siguientes tutoriales:

- NetBeans IDE Quick Start Guide: Tutorial para comenzar a trabajar con Netbeans.
- GUI Building in NetBeans IDE 5.5: Tutorial para aprender los conceptos básicos para crear el interfaz gráfico de una aplicación (GUI).

En la página web [http://www.netbeans.org/download/flash/netbeans\\_55/nb\\_overview/NetBeansOverview.html](http://www.netbeans.org/download/flash/netbeans_55/nb_overview/NetBeansOverview.html) puedes encontrar una breve descripción de cada uno de los componentes de la aplicación Netbeans 5.5, que nos indica para lo que sirve cada parte, cuando situamos el cursor del ratón sobre tal parte.

### 3.1. Importar código Java en netbeans

En esta sección vamos a ver cómo compilar y ejecutar, usando el entorno Netbeans, programas que ya teníamos construidos previamente sin usar el entorno Netbeans. O sea, vamos a ver cómo *Importar un proyecto desarrollado con otro entorno*. Para ello tendremos que crear un proyecto de la categoría *General* y de tipo *Java Project with Existing Sources*.

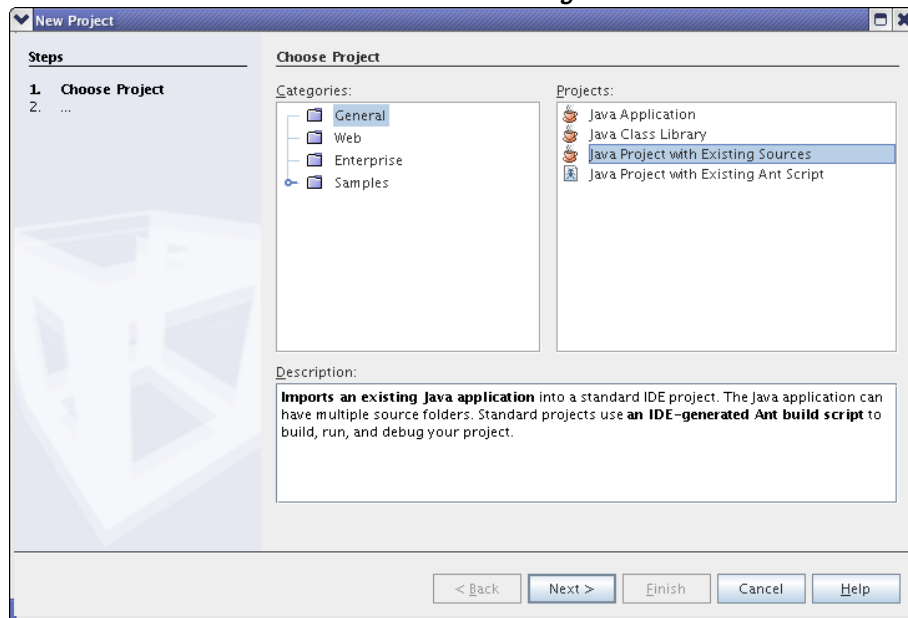


Cuando importamos un proyecto lo que ocurre es que el IDE crea un puntero a las carpetas que contienen su código fuente (no es copiado de nuevo) y se crea una carpeta que contiene metadatos para el proyecto.

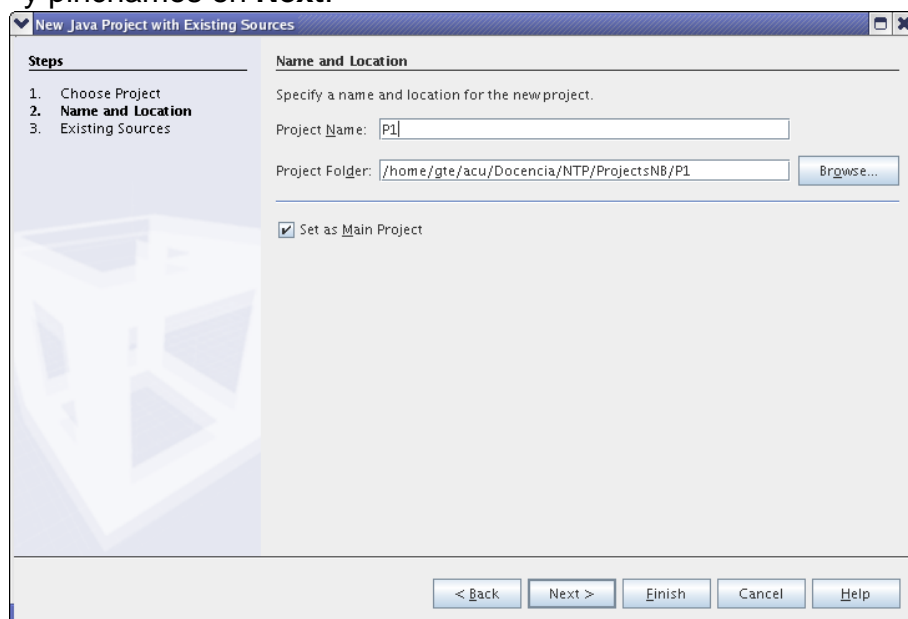
### 3.1.1. Programa Hello World

Vamos a compilar y ejecutar en primer lugar el programa `Example.java`. Copiamos el directorio (`/fenix/depar/ccia/ntp/Java/Sources/P1`) dónde está el programa `Example.java`, si no lo habíamos hecho ya, a alguna carpeta de vuestra cuenta. Si ya lo tenéis compilado con `javac`, conviene que borréis el fichero `.class`. A continuación realizamos los siguientes pasos:

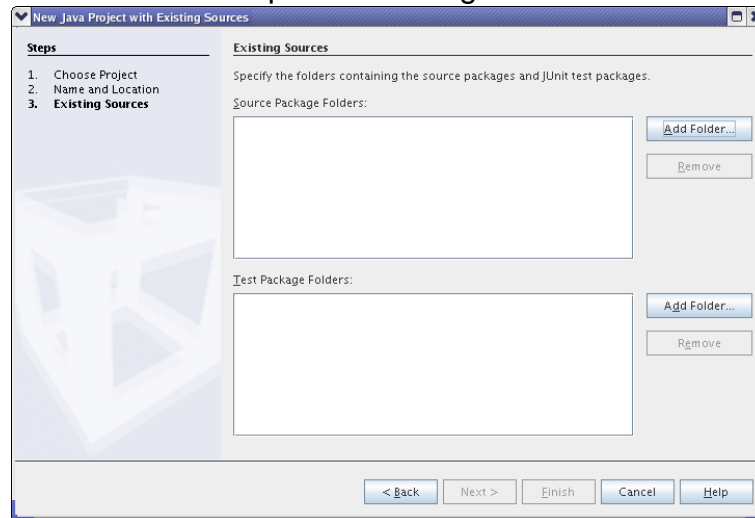
- En primer lugar debemos crear un nuevo proyecto. Para ello seleccionamos Menú `File` -> `New Project` (`Ctrl-Shift-N`)



- Seleccionamos `General` > `Java Project with Existing Sources` y pinchamos en **Next**.

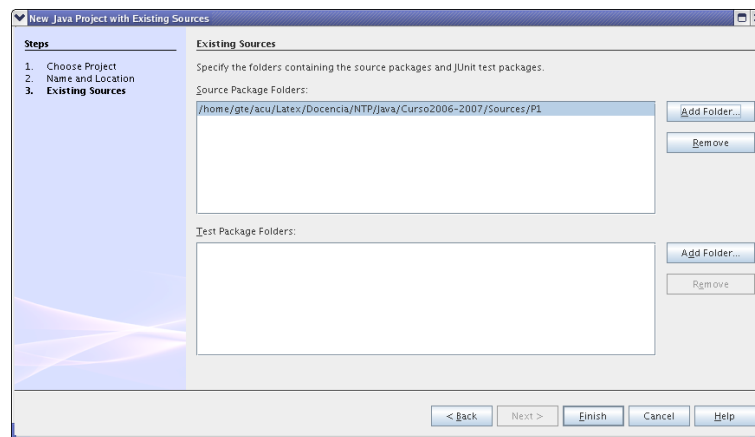


- Como nombre del proyecto introducimos P1. Como carpeta (directorio) donde colocar el proyecto usaremos el directorio `ProjectsNB/P1`. La carpeta `ProjectsNB` debe estar previamente creada dentro de tu *home*. Para ayudarte a seleccionar esta carpeta puedes pinchar en el botón **Browse**.
- Pulsa el botón **Next**. Aparecerá la siguiente ventana:

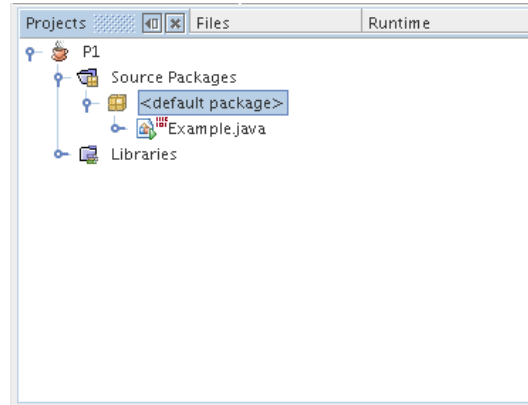


- Esta última ventana nos permite seleccionar la carpeta que contiene el código fuente de nuestro programa pinchando en el botón **Add Folder**. En mi caso he elegido la carpeta:

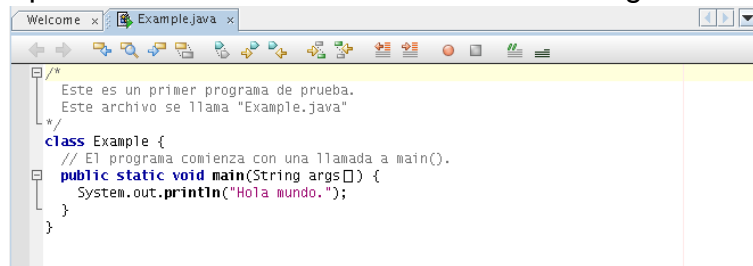
```
/home/gte/acu/Latex/Docencia/NTP/Java/Curso2006-2007/Sources/P1
```



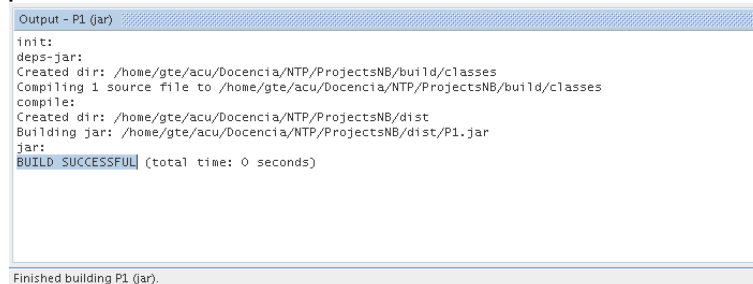
- Pulsa el botón **Finish** en la ventana anterior.
- El proyecto creado puede verse tanto en la ventana **Projects** como en la ventana **Files**:



- Pinchando en el icono de la izquierda (con forma de lupa horizontal), podemos abrir el directorio para ver los ficheros y directorios que contiene.
- Hacer doble click en el fichero `Example.java` de la ventana **Projects** para abrir este fichero en el editor del código fuente.



- Para **compilar** elegimos Menú **Build**→**Build Main Project** o bien pinchando con el botón derecho del ratón sobre el proyecto **P1** (en la ventana **Projects**), elegimos **Build Project**. Esto hará que en la ventana **Output Window**, colocada en la parte inferior derecha, aparezca el mensaje **BUILD SUCCESSFUL** indicando que la compilación terminó satisfactoriamente.



- Para **ejecutar** elegimos **Run**→**Run Main Project** o bien pinchando con el botón derecho del ratón sobre el proyecto **P1** (en la ventana **Projects**), elegimos **Run Project**. Esto hará que en la ventana **Output Window** aparezca el mensaje **Hola mundo** que es la salida que genera nuestro programa.

```

Output - P1 (run)
init:
deps-jar:
compile:
run:
Hola mundo.
BUILD SUCCESSFUL (total time: 0 seconds)

Finished building P1 (run).

```

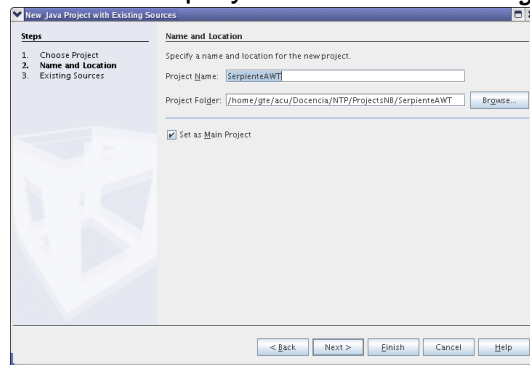
### 3.1.2. Programa de la serpiente

Compilemos y ejecutemos ahora el programa serpiente. Copiamos el directorio dónde está el programa, si no lo habíamos hecho ya, o sea la carpeta:

`/fenix/depar/ccia/ntp/Java/Sources/serpienteAWT.`

Si ya lo tenáis compilado con `javac`, conviene que borrréis todos los ficheros `.class`. A continuación realizamos los siguientes pasos:

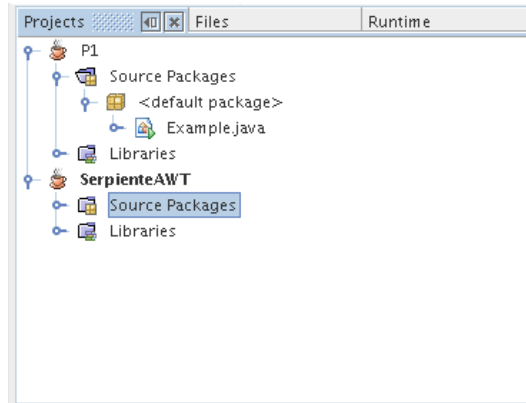
- Crear un nuevo proyecto. Para ello seleccionamos Menú File -> New Project (Ctrl-Shift-N)
- Seleccionar General > Java Project with Existing Sources y pinchamos en **Next**.
- Como nombre del proyecto introducimos `SerpienteAWT`. Como carpeta donde colocar el proyecto usaremos `ProjectsNB/SerpienteAWT`.



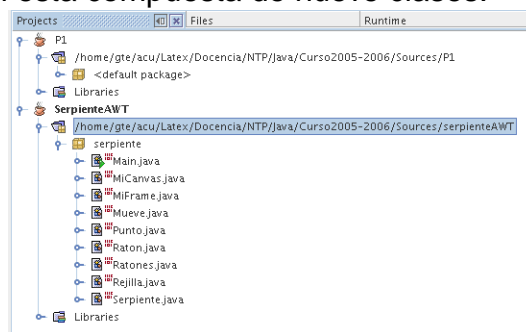
- Pulsa el botón **Next**.
- Añadir la carpeta `serpienteAWT` en la que tenemos el código fuente de la aplicación. En mi caso es:

`home/gte/acu/Latex/Docencia/NTP/Java/Curso2006-2007/Sources/serpienteAWT`

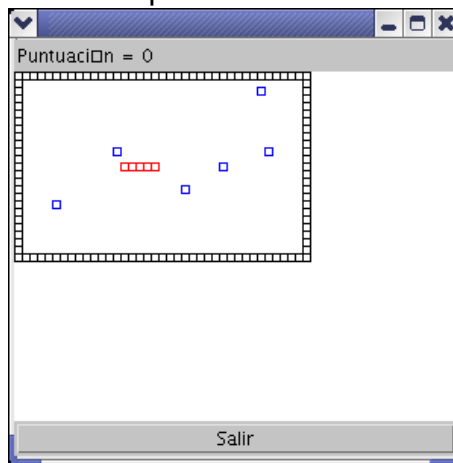
- Pulsa el botón **Finish** en la ventana anterior.
- El proyecto creado puede verse tanto en la ventana **Projects** como en la ventana **Files**:



- Expandiendo los iconos de este proyecto podemos comprobar que la aplicación está compuesta de nueve clases:



- Para **compilar** pinchamos con el botón derecho del ratón sobre el proyecto **SerpienteAWT** (en la ventana **Projects**), y seleccionamos **Build Project**. Esto hará que en la ventana **Output Window**, que está colocada en la parte inferior izquierda del IDE, aparezca el mensaje **BUILD SUCCESSFUL** indicando que la compilación terminó satisfactoriamente.
- Para **ejecutar** pinchamos con el botón derecho del ratón sobre el proyecto **serpienteAWT** (en la ventana **Projects**), y seleccionamos **Run Project**. Esto hará que aparezca la ventana **Frame** que muestra la aplicación de la serpiente.



### 3.2. Importing Existing Java Source Code Into NetBeans IDE 5.5

Puedes realizar ahora un tutorial similar disponible en la página [http://www.netbeans.org/kb/55/import\\_j2se.html](http://www.netbeans.org/kb/55/import_j2se.html)

## 4. Tutoriales en la página de Netbeans

### 4.1. NetBeans IDE 5.5 Quick Start Guide

Ahora realiza paso a paso el tutorial que puedes encontrar en la página <http://www.netbeans.org/kb/55/quickstart.html> que crea una nueva y simple aplicación desde el principio usando Netbeans.

### 4.2. Introduction to Developing General Java Applications

Realiza también el tutorial introductorio que se encuentra en la página <http://www.netbeans.org/kb/55/javase-intro.html>. Este tutorial enseña algunos conceptos bastante útiles tales como la característica de *completar código*, la *refactorización*, cómo definir los argumentos de entrada a `main()`, generación de documentación *javadoc*, cómo crear y ejecutar tests *JUnit*, introducción al uso del *depurador*, etc.